

**Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3

**«Создание таблиц базы данных PostgreSQL. Заполнение таблиц
рабочими данными»**

по дисциплине «Проектирование и реализация баз данных»

Обучающийся Баженов Алексей Антонович

Факультет прикладной информатики

Группа K3240

Направление подготовки 09.03.03 Прикладная информатика

Образовательная программа Мобильные и сетевые технологии 2025

Преподаватель Говорова Марина Михайловна

Санкт-Петербург

2025/2026

1. Цель работы:

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 1X, заполнения их рабочими данными, резервного копирования и восстановления БД.

2. Практическое задание:

- 1) Создать базу данных с использованием pgAdmin 4 (согласно индивидуальному заданию).
- 2) Создать схему в составе базы данных.
- 3) Создать таблицы базы данных.
- 4) Установить ограничения на данные: Primary Key, Unique, Check, Foreign Key.
- 5) Заполнить таблицы БД рабочими данными.
- 6) Создать резервную копию БД.

Указание:

Создать две резервные копии:

- с расширением CUSTOM для восстановления БД;
 - с расширением PLAIN для листинга (в отчете);
 - при создании резервных копий БД настроить параметры Dump options для Type of objects и Queries .
- 7) Восстановить БД.

3. Индивидуальное задание:

Вариант 7. БД «Курсы» (допустимо к выполнению для команды из 2-х студентов)

Описание предметной области: Сеть учебных подразделений НОУ ДПО занимается организацией внебюджетного образования.

Имеется несколько образовательных программ краткосрочных курсов, предназначенных для определенных специальностей, связанных с программным обеспечением ИТ. Каждая программа имеет определенную длительность (в академических часах), свой перечень изучаемых дисциплин, вид итоговой аттестации, вид документа об окончании программы (сертификат о повышении квалификации, удостоверение о повышении квалификации, диплом о профпереподготовке).

Одна дисциплина может относиться к нескольким программам.

На каждую программу может быть набрано несколько групп обучающихся.

По каждой дисциплине могут проводиться лекционные, лабораторные/практические занятия и практика в определенном объеме часов. По каждой дисциплине и практике проводится аттестация в формате экзамен/дифзачет/зачет.

Необходимо хранить информацию по аттестации обучающихся.

Подразделение обеспечивает следующие ресурсы: учебные классы, лекционные аудитории и преподавателей. Необходимо составить расписание занятий.

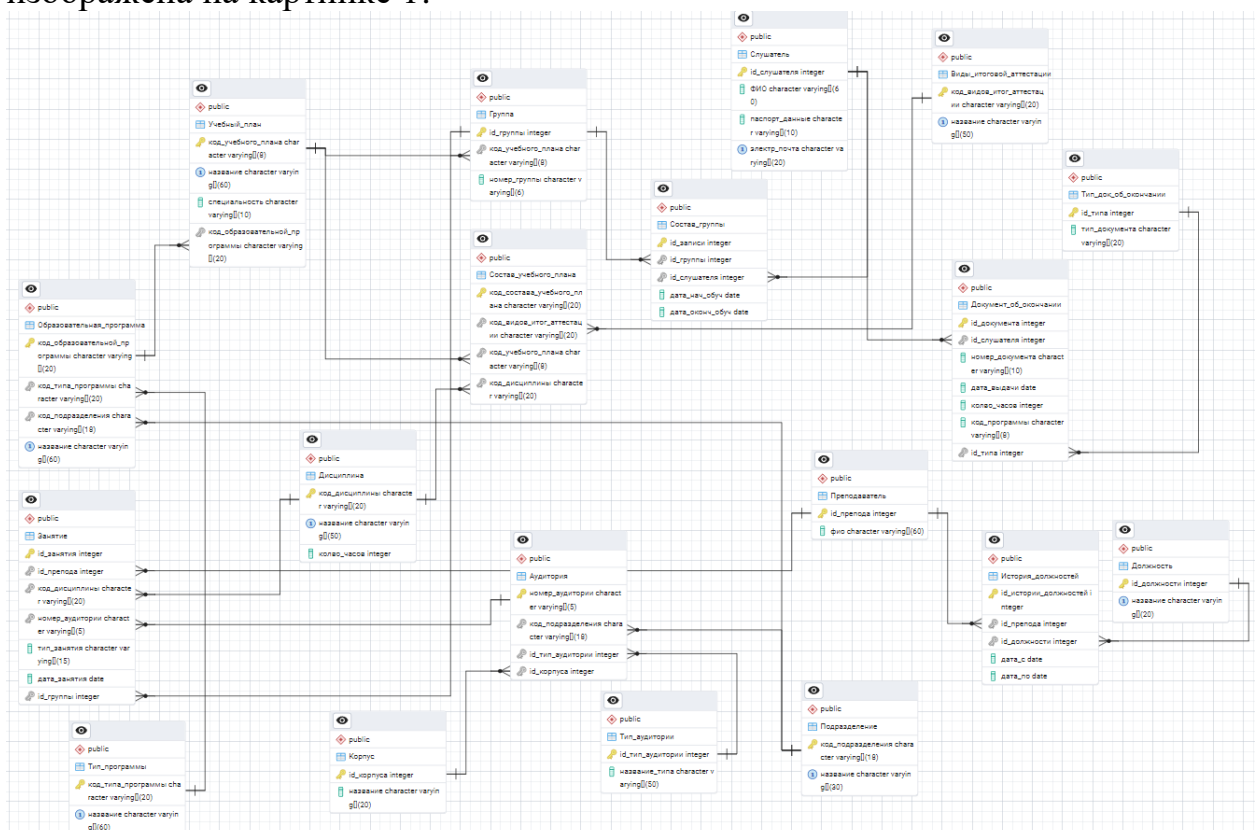
В системе необходимо хранить информацию о номере выданного документа об окончании программы и дату выдачи.

БД должна содержать следующий минимальный набор сведений: Фамилия слушателя. Имя слушателя. Паспортные данные. Контакты. Код программы. Программа. Тип программы. Объем часов. Номер группы. максимальное количество человек в группе (для набора). Дата начала обучения. Дата окончания обучения. Название дисциплины. Количество часов. Дата занятий. Номер пары. Номер аудитории. Тип аудитории. Адрес площадки. Вид занятий (лекционные, практические или лабораторные). Фамилия преподавателя. Имя и отчество преподавателя. Должность преподавателя. Дисциплины, которые может вести преподаватель.

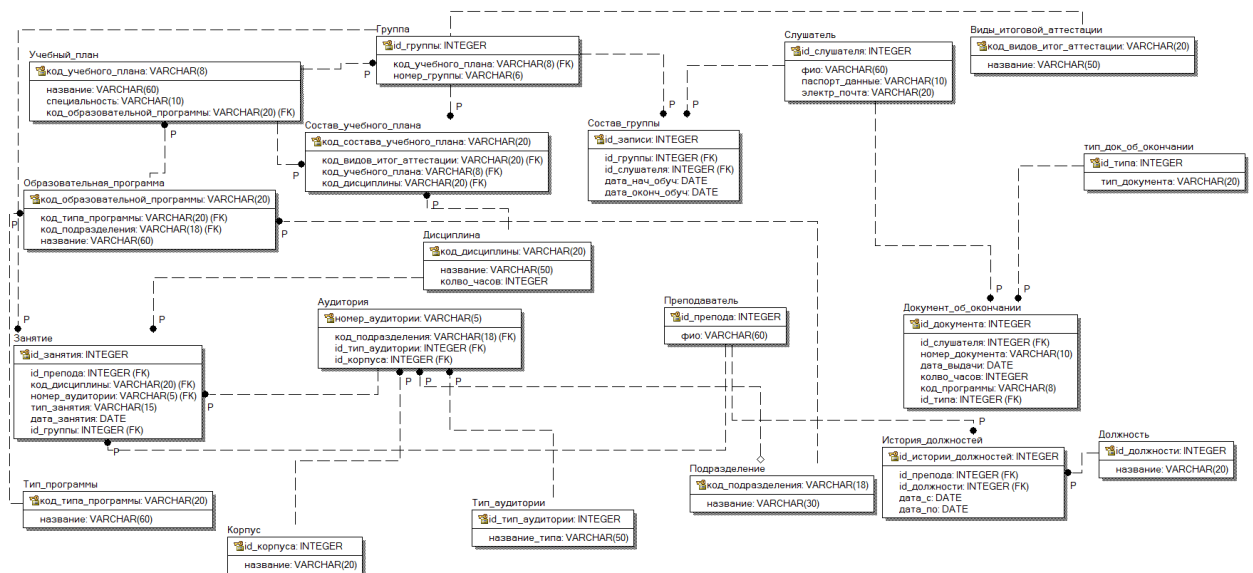
Дополните состав атрибутов на основе анализа предметной области.

4. Выполнение:

1. Наименование БД: courses
2. Схема логической модели базы данных, сгенерированная в Generate ERD изображена на картинке 1:



Картинка 1 - Схема ИЛМ, сгенерированная в Generate ERD



Картинка 2 - Схема ИЛМ в нотации IDEF1X

3. Работа с резервными копиями БД.

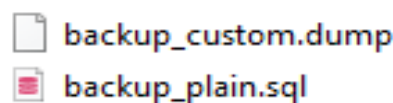
- Создание резервной копии с расширением .custom:

```
# -U postgres → подключение под пользователем postgres
# -F c → формат дампа "custom" (сжатый, двоичный)
# -f backup_custom.dump → имя выходного файла дампа
# --create → включает команду CREATE DATABASE в дампе
pg_dump -U postgres -F c -f backup_custom.dump courses --create
```

- Создание дампа базы данных courses в формате PLAIN (обычный SQL)

```
# -F p → формат plain (текстовый .sql)
# -f backup_plain.sql → имя выходного файла
# --create → включает команду CREATE DATABASE
pg_dump -U postgres -F p -f backup_plain.sql courses --create
```

Итоговые файлы представлены на картинке 3:



Картинка 3 – Резервные копии баз данных

- Восстановление из .custom:

```
# Создание новой базы данных, в которую будет загружен дампе
createdb -U postgres restored_db
# Восстановление базы данных из дампа формата CUSTOM
# -d restored_db → целевая база данных
# -v → подробный (verbose) режим
pg_restore -U postgres -d restored_db -v backup_custom.dump
```

- Восстановление из .sql:

```
# -d restored_db → база, в которую будет загружен SQL  
psql -U postgres -d restored_db -f backup_plain.sql
```

4. Вывод

В ходе выполнения лабораторной работы были достигнуты следующие цели:

- Созданы таблицы, включающих:
 - Определение первичных и внешних ключей;
 - Ограничения целостности данных (NOT NULL, CHECK, FOREIGN KEY).
- Таблицы были успешно заполнены тестовыми рабочими данными при помощи SQL-запросов INSERT INTO.
- Проверена корректность структуры базы данных и целостность данных с помощью запросов SELECT.

В результате выполнения лабораторной работы были закреплены практические навыки проектирования и создания таблиц в PostgreSQL, определения связей между таблицами, а также первичной и тестовой загрузки данных.